GC-F-004v.08

Análisis y desarrollo de software

FASE 3 Ejecución – GA7-AA3

Octubre 14 de 2025

@SENAComunica

www.sena.edu.co

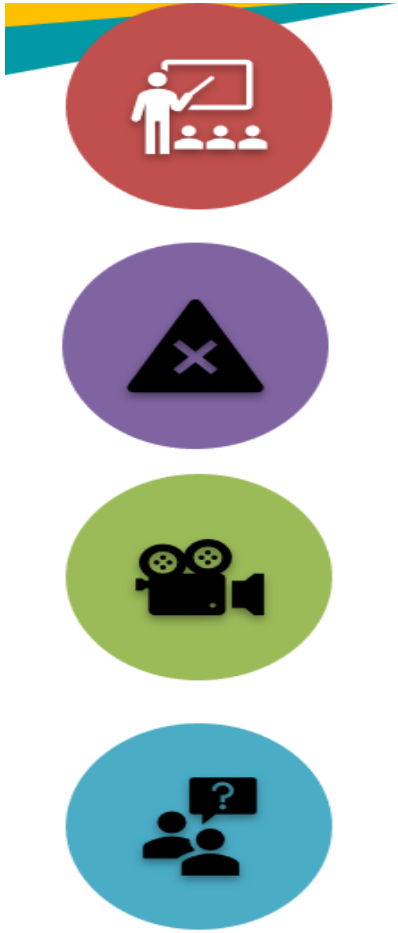
Agenda del día

1. Saludo de Bienvenida.
2. Recomendaciones Generales
3. Evidencias a desarrollar

✓**Evidencia de desempeño: GA7-220501096-AA3-EV01 codificación de módulos del software Stand alone, web y móvil de acuerdo al proyecto a desarrollar .**

4. Resolución de Dudas e Inquietudes
5. Despedida Equipo de Instructores

Recomendaciones generales



1. Estudiar el material de la semana antes de la videoconferencia.
2. Evita distractores durante la sesión.
3. Toma nota de las dudas y preguntas que surjan durante la sesión y ten presente las que te hayan surgido durante el estudio del material.
4. Al final de la sesión, se asignarán 10 minutos para resolver todas las inquietudes que se hayan presentado.

3.1.3 Actividad de aprendizaje GA7-220501096-AA3 - Codificar los módulos del software Stand-alone, web y móvil

Esta actividad se centra en la codificación del módulo del proyecto según las características del software a desarrollar, utilizando Frameworks de Java para el desarrollo ágil.

Duración: 122 horas

Materiales de formación a consultar: para el desarrollo de esta actividad es importante la lectura y análisis del componente formativo: "**Frameworks para construcción de aplicaciones con JAVA**".

Evidencias: a continuación, se describen las acciones y las correspondientes evidencias que conforman la actividad de aprendizaje**:**

Evidencia de desempeño: GA7-220501096-AA3-EV01 codificación de módulos del software Stand alone, web y móvil de acuerdo al proyecto a desarrollar

Con base en la selección del proyecto a desarrollar móvil o web realice la codificación del módulo del proyecto aplicando alguno de los framework vistos en el componente formativo “Frameworks para construcción de aplicaciones con JAVA.”.

Elementos para tener en cuenta:

- Para la codificación del módulo debe tener en cuenta los artefactos del ciclo del software realizados con anterioridad: diagrama de clases, diagramas de casos de uso, historias de usuario, diseños, prototipos, Informe técnico de plan de trabajo para construcción de software con tecnologías seleccionadas etc**.**
- El código debe contener comentarios
- El código debe cumplir con estándares de codificación
- Se debe crear el proyecto utilizando herramientas de versionamiento.

Lineamientos generales para la entrega de la evidencia:

- **Productos para entregar:** carpeta comprimida que debe tener los siguientes archivos: ○ archivos del proyecto, ○ archivo con enlace del repositorio, la carpeta comprimida debe tener el nombre del aprendiz y número de la evidencia así: NOMBRE_APELLIDO_AA3_EV01
- **Extensión:** ZIP, RAR.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio:

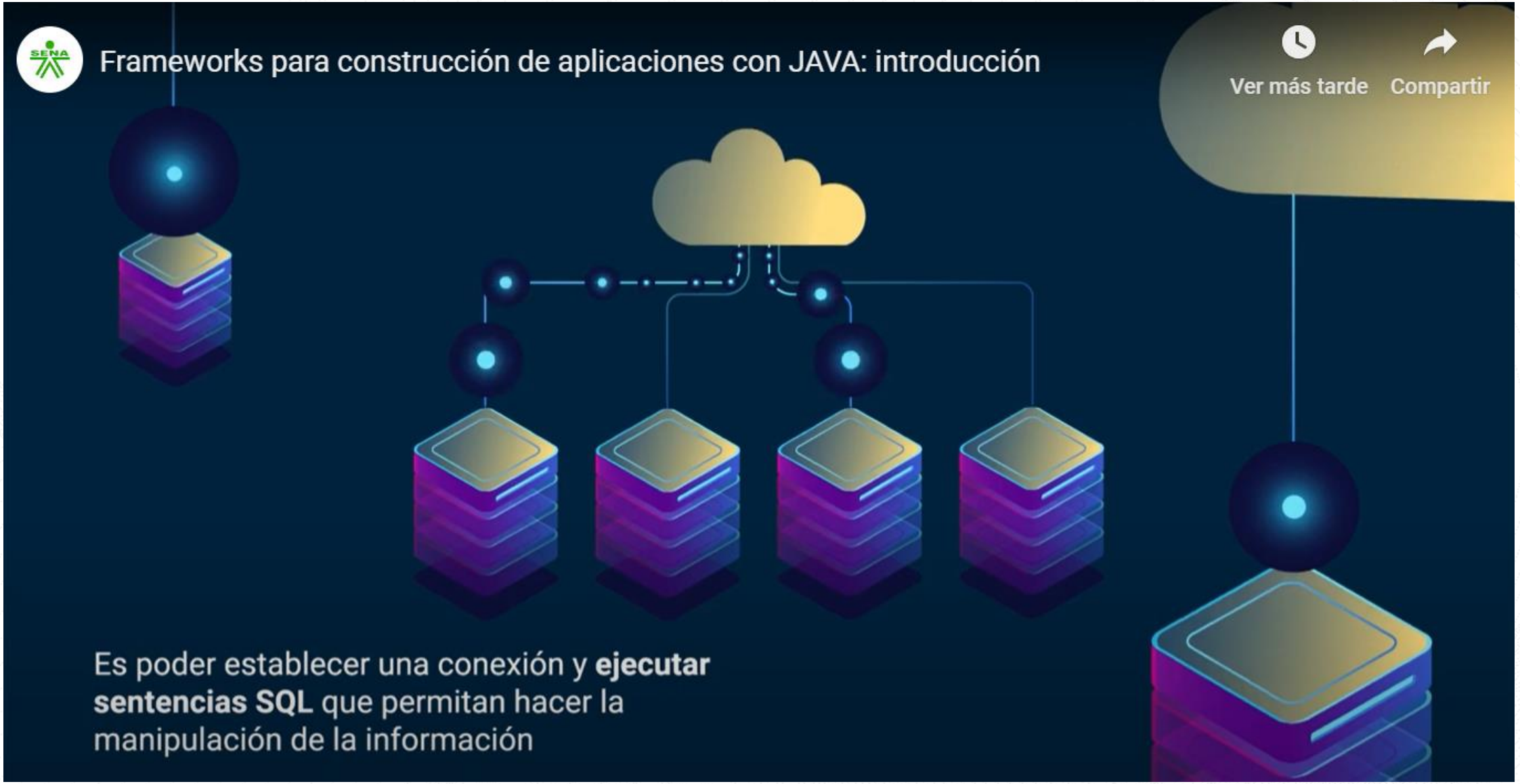
codificación de módulos del software Stand alone, web y móvil de acuerdo al proyecto a desarrollar GA7220501096-AA3-EV01.

Criterios de evaluación

• Codifica los módulos del software Stand alone, web y móvil, de acuerdo con las especificaciones del diseño y el estándar de codificación.

2. LISTA DE CHEQUEO

No.	VARIABLES/INDICADORES DE LOGRO	CUMPLE		Observaciones
		SI	NO	
1.	Selecciona y aplica un Framework para la codificación del proyecto.			30%
2.	Integra herramientas para almacenamiento de datos			30%
3.	Utiliza un estándar de codificación definido y usa comentarios en el código.	✓		30%
4.	Utiliza un estándar de codificación definido.			10%


Frameworks para construcción de aplicaciones con JAVA: introducción
Ver más tarde
Compartir
Es poder establecer una conexión y **ejecutar sentencias SQL** que permitan hacer la manipulación de la información



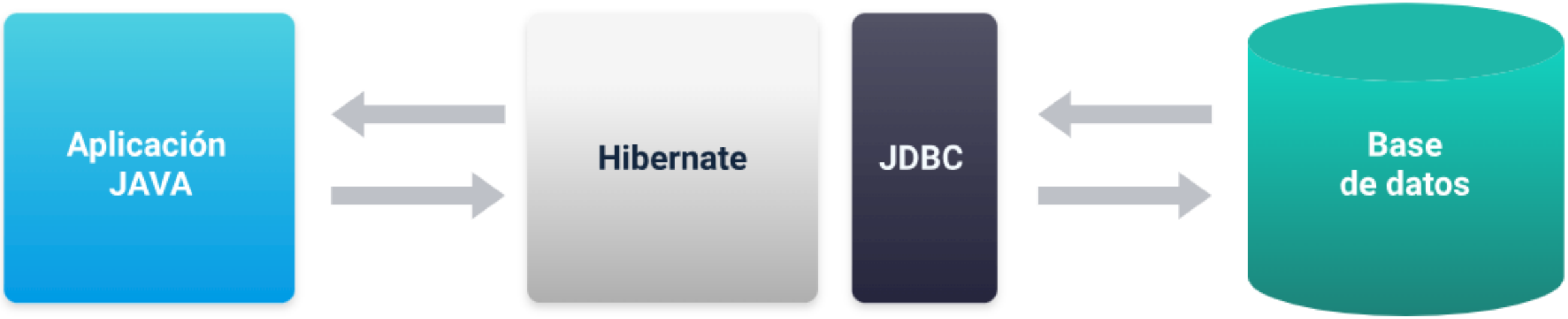
Aspecto	Bases de datos relacionales	Java (orientado a objetos)
Estructura	Tablas con filas y columnas	Clases con atributos y métodos
Relaciones	Mediante claves primarias y foráneas	Mediante referencias a objetos
Lenguaje de consulta	SQL	Métodos y expresiones del lenguaje
Tipo de datos	Tipos simples (INT, VARCHAR, DATE)	Tipos complejos (objetos, listas, etc.)
Persistencia	Permanente en disco	Temporal en memoria (a menos que se guarde en BD)
Herencia	No nativa (se simula con relaciones)	Soportada directamente
Manipulación de datos	Basada en tablas y registros	Basada en objetos y métodos



Hibernate es software libre con licencia GNU LGPL que presta servicios de ORM para JAVA, es decir, es plugin que puede ser integrado al IDE de desarrollo en JAVA que facilita el proceso de mapeo de tablas en una base de datos relacional a objetos JAVA mediante el uso de archivos declarativos en formato XML o anotaciones.

> *Hibernate* utiliza JDBC para todos los procesos de comunicación con la base de datos funcionando como una capa adicional de abstracción de las funcionalidades del JDBC. En una aplicación JAVA podremos delegar la mayoría de las operaciones de bases de datos al *framework* de *Hibernate* pero en segundo plano todo estará pasando por la API de JDBC.
>
> **Martinez (2015)**

Esquema general del funcionamiento de Hibernate

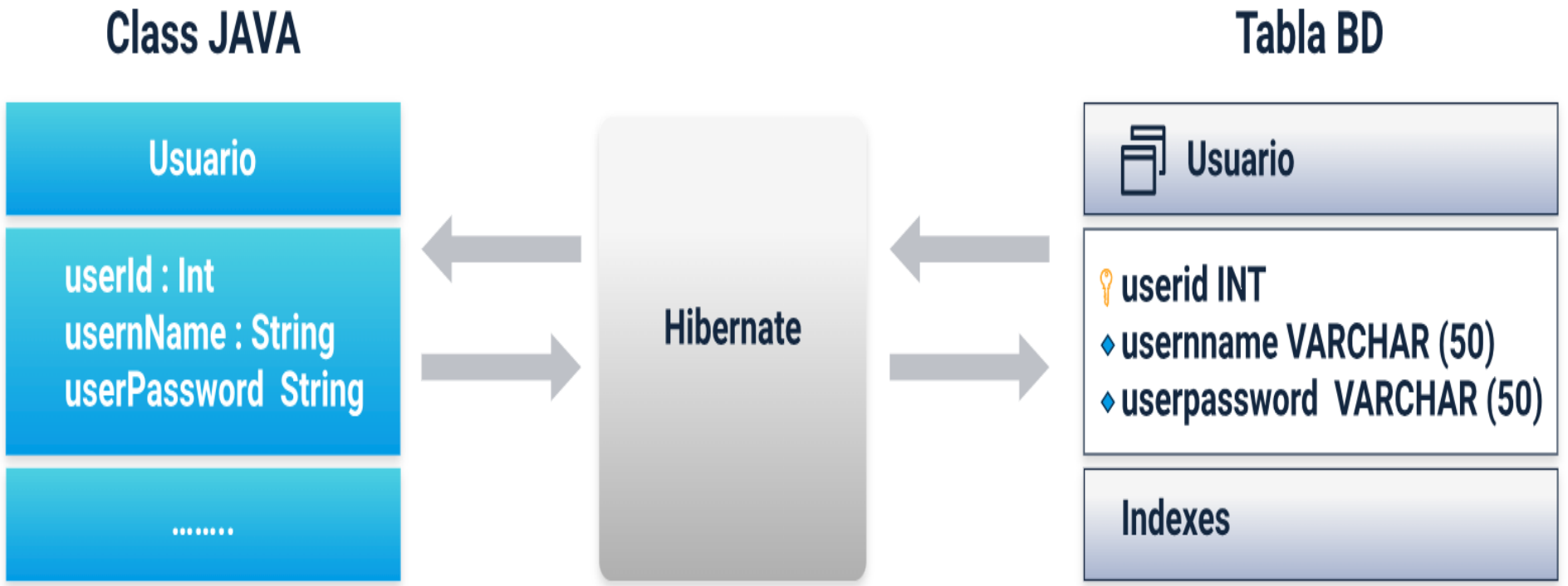


Para el desarrollo de una aplicación JAVA que usa el entorno de desarrollo de *Eclipse* es necesario descargar todas las librerías requeridas por el *framework* de *Hibernate* las cuales puede encontrar en la página oficial del ORM https://hibernate.org/orm/. Una vez descargado el paquete este se debe descomprimir y podrá encontrar los archivos jar necesarios en la ubicación /lib/required los cuales deberá anexar al proyecto JAVA.

Como instalar Hibernate en Netbeans latest version

https://www.youtube.com/watch?v=431fhzjUWf4

Esquema de mapeo entre una clase Java y una tabla de base de datos con Hibernate



CRUD Java Web con Hibernate, JSF y PimerFaces MySql Netbean

https://www.youtube.com/watch?v=R2UjK9TcYMw

https://mega.nz/file/35BkESrK#Ud4FSAUjvenmOxRdMa2o31xc1JAyYGW9GnVJ-nkmkbE



Es un *Framework* de código abierto que facilita crear aplicaciones empresariales de alto rendimiento en JAVA, además de JAVA tiene soporte para los lenguajes *Groovy* y *Kotlin*. La estructura que lo compone es modular esta estructura cuenta con una flexibilidad para implementar una variedad de arquitecturas dependiendo de los requerimientos de las aplicaciones haciendo que la programación sea lo más hacedera posible, rápida y muy segura.

> Historia: Rod Johnson creador de *Spring* se da cuenta que necesitan una herramienta para simplificar y facilitar la construcción de aplicaciones JEE utilizando EJB y escribe el libro expert *one-on-one* J2EE *Development* que es un conjunto de buenas prácticas para la creación de aplicaciones empresariales y en el año 2004 lanza primera versión 1.2.6 de *Spring*, según el autor era el inicio de J2EE tradicional.
>
> **Johnson, (2004)**

Unas de las características principales de *Spring* son:

1. Flexibilidad
2. Inyección de dependencia (DI)
3. POJOs (Plan *Old Java Objects*):
4. ORM (*Object Relational Mapping*)
5. **Estructura y módulos**

   Spring integra una variedad de módulos que se pueden integrar para el desarrollo de las aplicaciones.



La siguiente figura nos muestra lo que se puede crear con *Spring Framework*: Aplicaciones de consola, Aplicaciones de escritorio, Aplicaciones Web, Aplicaciones Móviles, Servicios SOAP, REST. *Spring* está orientado esencialmente al *backend*.



Microservicios



Nube



Aplicaciones web



Reactive



Servlets



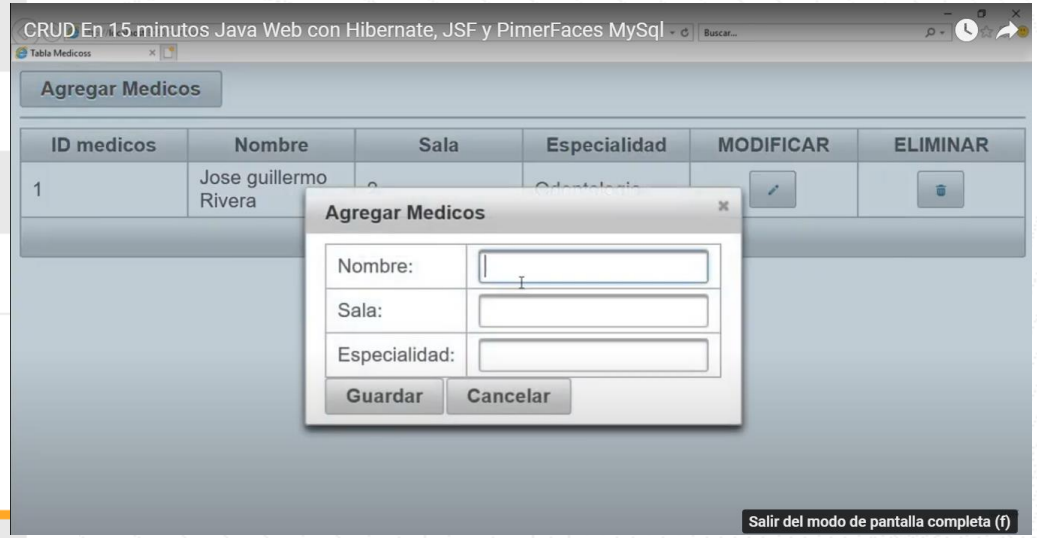
Batch

Tabla 1. Entornos para desarrollo en *Spring*

Entornos de desarrollo	Enlace
IntelliJ IDEA	https://www.jetbrains.com/idea/
Spring Tools	https://spring.io/tools
Visual Studio Code	https://code.visualstudio.com/docs/languages/java
Eclipse	https://www.eclipse.org/downloads/packages/
NetBeans	https://netbeans.apache.org/

Tabla 2. JDK para desarrollo en *Spring*

JDK	Enlace
JDK Oracle	https://www.oracle.com/java/technologies/downloads/
AdoptOpenJDK	https://adoptopenjdk.net/



Plazo Actividades:

14 al 26 de Octubre



G R A C I A S

Línea de atención al ciudadano: 01 8000 910270
Línea de atención al empresario: 01 8000 910682

@SENAComunica

www.sena.edu.co